UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/727,393	11/29/2000	Allen P. Chen	10559-385001/P10191	4146

20985 7590 06/27/2007
FISH & RICHARDSON, PC
P.O. BOX 1022
MINNEAPOLIS, MN 55440-1022

EXAMINER	
SEFCHECK, GREGORY B	

ART UNIT	PAPER NUMBER
2616	

MAIL DATE	DELIVERY MODE
06/27/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

U
ST

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

**BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES**

Application Number: 09/727,393
Filing Date: November 29, 2000
Appellant(s): CHEN ET AL.

MAILED
JUN 27 2007
GROUP 2600

Tonya Drake
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed 2/28/2007 appealing from the Final Office action mailed 7/17/2006 and Pre-Appeal decision mailed 1/16/2007.

(1) Real Party in Interest

A statement identifying by name the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

The statement of the status of claims contained in the brief is correct.

(4) Status of Amendments After Final

No amendment after final has been filed.

(5) Summary of Claimed Subject Matter

The summary of claimed subject matter contained in the brief is correct.

(6) Grounds of Rejection to be Reviewed on Appeal

The appellant's statement of the grounds of rejection to be reviewed is correct.

(7) Claims Appendix

The copy of the appealed claims contained in the Appendix to the brief is correct.

(8) Evidence Relied Upon

6,671,758	CAM et al.	12-2003
5,440,545	BUCHOLZ et al.	8-1995
6,144,622	COLMANT et al.	11-2000
6,847,644	JHA	1-2005
6,105,122	MULLER et al.	8-2000

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

Claim Rejections - 35 USC § 103

1. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

2. Claims 1-4, 6, 7, 9-11, 13, 15, 16, 18, 19, 24, 26-33, 35-42 are rejected under 35 U.S.C. 103(a) as being unpatentable over Cam et al. (US006671758B1), hereafter Cam, in view of Bucholz et al. (US005440545A), hereafter Bucholz.

- In regards to Claims 1, 2, 4, 6, 9, 10, 13, 15, 16, 18, 19, 24, 26-28, 32, 35-41,

Cam discloses a packet data transfer method on an interface having a large number of ports (Abstract; claim 1,9,24 – intra-packet switching).

Referring to Fig. Cam shows that a Layer/master device 22 polls the PHY devices 14-20 to determine which have data waiting to be transferred (Col. 1, lines 38-41; claim 1,9,24,32 – determining which ports contain a data packet available for processing).

Cam shows that data packets waiting to be transferred from polled PHY devices are fragmented to a maximum block size (cell) of data. This maximum block size may be fixed at start-up or by programming through an external management interface (Col. 3, lines 2-9; claim 1,9,24,32 – fragmenting available data packet into at least one cell having defined size; claim 1,9,24,26,32 – fragmentation continues until a user-defined number of cells are generated; claim 2,10,27 – monitoring the number of cells produced to determine if user-defined number are generated; claim 18,19 – user interface for allowing user to specify user-defined cells to be generated by packet fragmentation process).

Referring to Figs. 5, 6, and 11, Cam shows that packets awaiting transfer from multiple PHY devices will be transferred a maximum block size at a time before deselecting itself, at which point the next PHY device begins transfer of a maximum block size. As such, a first selected PHY device having a first packet to be transferred that is greater than the maximum block size (i.e. 256 bytes; Col. 3, lines 6-7) will fragment and transmit a first portion of the packet during a first transfer period. The selected PHY is deselected after transmission of the maximum block size, a subsequent PHY device is selected and packet fragmentation and transmission is repeated for that newly-selected PHY device, up to the maximum block size per transfer period. The

remaining portions of the first packet at the first PHY device will be fragmented and transferred during the next selected transfer period, after all other PHY devices have been given a chance to transmit (Col. 3, lines 6-7 and line 55-Col. 4, line 24; claim 1,9,24,26,32 - fragmenting a first portion of a second available data packet on a different port subsequent to fragmenting a first portion of the first data packet but prior to fragmenting a second portion of the first available data packet; claim 40 – user-defined number of cells comprises at least 2 cells).

Cam does not explicitly disclose fragmentation of the first available packet using a signal processing circuit and storing, in memory of the processing circuit, a data element indicative of the incomplete fragmentation status concerning the first available packet, comprising a data packet length remainder indicative of the packet portion not fragmented and a packet truncation indicator indicative of incomplete fragmentation of the packet, if another port contains an available packet, where the element allows subsequent processing of the remainder of the data packet, after subsequent fragmentation of a second packet.

Bucholz discloses a packet delivery system in which packets are fragmented for transmission (Title; Abstract). Referring to Fig. 6, Bucholz shows that a reassembly header (stored data element) is stored in the fragmented packet indicating its place within the packet, total packet length, total fragments, etc. such that the progress of the packet's fragmentation can be determined (Col. 6-7, lines 63-23; claim 1,4,9,24,32 – storing, in memory, data element indicative of the incomplete fragmentation status

concerning the packet being processed if another port contains an available packet, where the element allows subsequent process of the remainder of the data packet being processed; claim 13,35,37 – data packet length remainder indicative of packet portion not fragmented; claim 13 – packet truncation indicator indicative of incomplete fragmentation of the packet; claim 6,15,16,28 – monitoring and determining if the data packet has been fully fragmented; claim 36 – indicator indicative of the length of a packet previously fragmented; claim 36,37 - indicator indicative of total length of packet; claim 38 – fragmenting a second portion of the first packet subsequent to fragmenting a first portion of a second packet; claim 39 – using the stored data element to enable fragmenting the second portion of the first packet).

Referring to Figs. 1b and 2, Bucholz discloses that packet fragmentation is performed in a packet switching device 100 that comprises a processor 110 (signal processing circuit) and associated memory 232 for storing the reassembly header (claim 41 - fragmentation of the first available packet using a signal processing circuit; claim 41 – storing data element concerning the first available packet in memory of the processing circuit).

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify the method of Cam by processing packet fragments in a signal processing circuit and storing a reassembly header for a first packet being processed in memory of the processing circuit, the header including information regarding total packet length, how much of the packet has been fragmented and how much remains to be fragmented, so that subsequent fragment processing of other packets and the

remainder of the first packet can be performed, as taught by Bucholz. This would enable determination of when packet fragmentation of a large packet transmitted in multiple fragments has been completed, as well as enable the transmission system of Cam to recognize the progress and status of packet fragmentation so that packets need not be completely fragmented before the per-device polling mechanism proceeds to processing of packets at other PHY devices.

- In regards to Claims 3 and 11,

Cam discloses a packet data transfer process that covers all limitations of the parent claim.

Referring to Fig. 5, Cam shows that polling of the PHY devices to determine if packets are available for processing is continuously done so that it is determined if any other ports contain packets available for processing other than those that have already so indicated (Col. 1, lines 38-41; Col. 3, lines 19-24; claim 3,11 – re-determining if packets are available for processing on any of the plurality of ports if the number of cells have been generated from the first port determined to have a packet available).

- In regards to Claim 7,

Cam discloses a packet data transfer process that covers all limitations of the parent claim.

Referring to Fig. 5, Cam shows that polling of the PHY devices to determine if packets are available for processing is continuously done so that it is determined if any

other ports contain packets available for processing other than those that have already so indicated (Col. 1, lines 38-41; Col. 3, lines 19-24; claim 7 – re-determining if packets are available for processing on any of the plurality of ports if the number of cells have been generated from the first port determined to have a packet available).

- In regards to Claims 29-31 and 33,

Cam discloses a packet data transfer process that covers all limitations of claim 29 similar to claims 1, 9, 24, and 32 as shown above.

Cam does not explicitly show implementing the methods and processes through computer programs residing on computer readable medium such as read-only and random access memory in which the processor and memory are on a single board computer.

It is well known that software implementation by a computer having a processor and memory for performing process/method functions can be cost efficient and enable accessibility for updates/upgrades of the processes to accommodate new technologies (claim 29 – computer program product residing on computer readable medium; claim 30 – computer readable medium is ROM; claim 31 - computer readable medium is RAM; claim 33 – processor and memory are incorporated into single board computer).

It would have been obvious to one of ordinary skill in the art at the time of the invention to implement the method of Cam through a processor running software from a read-only or random access memory in a single board computer. Implementations of methods and processes through software instructions residing on computer readable

medium such as ROM and RAM can be much less expensive than hardware implementations and provide accessibility for updates/upgrades of the processes to accommodate new technologies.

- In regards to Claim 42,

Cam discloses a packet data transfer method on an interface having a large number of ports (Abstract; claim 42 – programmable intra-packet switching method).

Referring to Fig. Cam shows that a Layer/master device 22 polls the PHY devices 14-20 to determine which have data waiting to be transferred (Col. 1, lines 38-41; claim 42 – determining which ports contain a data packet available for processing).

Cam shows that data packets waiting to be transferred from polled PHY devices are fragmented to a maximum block size (cell) of data. This maximum block size may be fixed at start-up or by programming through an external management interface (Col. 3, lines 2-9; claim 42 – fragmenting available data packet into at least one cell having defined size; claim 42 – fragmentation continues until a user-defined number of cells are generated; claim 42 – monitoring the number of cells produced to determine if user-defined number are generated).

Referring to Figs. 5, 6, and 11, Cam shows that packets awaiting transfer from multiple PHY devices will be transferred a maximum block size at a time before deselecting itself, at which point the next PHY device begins transfer of a maximum block size. As such, a first selected PHY device having a first packet to be transferred that is greater than the maximum block size (i.e. 256 bytes; Col. 3, lines 6-7) will

fragment and transmit a first portion of the packet during a first transfer period. The selected PHY is deselected after transmission of the maximum block size, a subsequent PHY device is selected and packet fragmentation and transmission is repeated for that newly-selected PHY device, up to the maximum block size per transfer period. The remaining portions of the first packet at the first PHY device will be fragmented and transferred during the next selected transfer period, after all other PHY devices have been given a chance to transmit (Col. 3, lines 6-7 and line 55-Col. 4, line 24; claim 42 – re-determining which ports contain a data packet available for processing if the user defined number of cells have been generated; claim 42 - fragmenting a first portion of a second available data packet on a different port subsequent to fragmenting a first portion of the first data packet but prior to fragmenting a second portion of the first available data packet).

Cam does not explicitly disclose storing, in memory, a data element including a first data element indicative of the incomplete fragmentation status concerning the first available packet and a second data element indicative of the packet portion not fragmented or remaining to be fragmented, if another port contains an available packet, where the data element allows subsequent processing of the remainder of the data packet, after subsequent fragmentation of a second packet on a different port.

Bucholz discloses a packet delivery system in which packets are fragmented for transmission (Title; Abstract). Referring to Fig. 6, Bucholz shows that a reassembly header (stored data element) is stored in the fragmented packet indicating its place

within the packet, total packet length, total fragments, etc. such that the progress of the packet's fragmentation can be determined (Col. 6-7, lines 63-23; claim 42 – storing, in memory, data element indicative of the incomplete fragmentation status concerning the packet being processed if another port contains an available packet, where the element enables subsequent process of the remainder of the data packet being processed; claim 42 – second data element indicative of packet portion not fragmented or length of a packet previously fragmented; claim 42 – data element enables fragmenting a second portion of the first packet subsequent to fragmenting a first portion of a second packet on a different port).

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify the method of Cam by processing packet fragments and storing a reassembly header for a first packet being processed including information regarding incomplete fragmentation status of the packet and either how much of the packet has been fragmented or how much remains to be fragmented, so that subsequent fragment processing of other packets on different ports as well as the remainder of the first packet can be performed, as taught by Bucholz. This would enable determination of when packet fragmentation of a large packet transmitted in multiple fragments has been completed, as well as enable the transmission system of Cam to recognize the progress and status of packet fragmentation so that packets need not be completely fragmented before the per-device polling mechanism proceeds to processing of packets at other PHY devices.

3. Claims 5, 8, 14, and 17 are rejected under 35 U.S.C. 103(a) as being unpatentable over Cam in view of Bucholz as applied to claims 4, 7, 11, and 16 above, and further in view of Colmant et al. (US006144662A).

- In regards to Claims 5, 8, 14, and 17,

Cam discloses a packet data transfer process that covers all limitations of the parent claim.

Cam does not explicitly show initiating fragmentation on a data packet from another port while the fragmentation of the data packet on the first port continues until user-defined cells are generated.

Colmant discloses a non-blocking switch. Referring to Figs. 1B and 3B, Colmant shows that a packet received on a port 85 may be fragmented into several portions (6001-6004) while another received packet on another port 85 is fragmented into portions (6125-6128), thereby enabling multiple port packet fragmentations to operate in parallel, thereby improving the utilization of the transmission medium and keeping waiting time for a blocked output port low (Col. 9, lines 14-42; claim 5,8,14,17 – initiating fragmentation on a data packet from another port while the fragmentation of the data packet on the first port continues until user-defined cells are generated).

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify the method of Cam to enable a received packet on another port proceed with fragmentation while the fragmentation of a first packet on the first port

continues. By fragmenting received packets in parallel, processing delay required to fragment a packet prior to transmission could be eliminated.

4. Claims 20-23 and 34 are rejected under 35 U.S.C. 103(a) as being unpatentable over Cam in view of Bucholz, as applied to claims 9 and 32 above, and further in view of Jha (US006847644B1).

- In regards to Claims 20-22 and 34,

Cam discloses a packet data transfer method on an interface having a large number of ports that covers all limitations of parent claim 9. (Abstract).

Referring to Fig. Cam shows that a Layer/master device 22 polls the PHY devices 14-20 to determine which have data waiting to be transferred (Col. 1, lines 38-41; claim 21 – determining which ports contain a data packet available for processing).

Cam shows that data packets waiting to be transferred from polled PHY devices are fragmented to a maximum block size (cell) of data. This maximum block size may be fixed at start-up or by programming through an external management interface (Col. 3, lines 2-9; claim 21 – fragmenting available data packet into at least one cell having defined size; claim 21 – fragmentation continues until a user-defined number of cells are generated; claim 22 – monitoring the number of cells produced to determine if user-defined number are generated).

Referring to Figs. 5, 6, and 11, Cam shows that packets awaiting transfer from multiple PHY devices will be transferred a maximum block size at a time before

deselecting itself, at which point the next PHY device begins transfer of a maximum block size. As such, a first selected PHY device having a first packet to be transferred that is greater than the maximum block size (i.e. 256 bytes; Col. 3, lines 6-7) will fragment and transmit a first portion of the packet during a first transfer period. The selected PHY is deselected after transmission of the maximum block size, a subsequent PHY device is selected and packet fragmentation and transmission is repeated for that newly-selected PHY device, up to the maximum block size per transfer period. The remaining portions of the first packet at the first PHY device will be fragmented and transferred during the next selected transfer period, after all other PHY devices have been given a chance to transmit (Col. 3, lines 6-7 and line 55-Col. 4, line 24; claim 21 - fragmenting a first portion of a second available data packet on a different port subsequent to fragmenting a first portion of the first data packet but prior to fragmenting a second portion of the first available data packet).

Cam does not explicitly show the plurality of ports connected to a synchronous optical network and the fragmentation process producing ATM cells. Cam also does not explicitly show a processor and memory of an ATM/POS processor for performing the method.

Jha discloses hybrid data transport over optical networks. Referring to Fig. 2 and 4, a switch's ports are connected to a SONET network. Jha shows that SONET data frames may be inverse-multiplexed (fragmented) into smaller bandwidth streams, such as ATM cell streams (Col. 1-2, lines 58-6; claim 21 - plurality of ports connected to a synchronous optical network; claim 20,21 - fragmentation process producing ATM

cells). Referring to Fig. 3, Jha also shows that processing of ATM and Packet-over-SONET can be accommodated through a single switch 102a/n (claim 34 – processor and memory are incorporated into ATM/POS processor).

It would have been obvious to one of ordinary skill in the art at the time of the invention to implement the method of Cam through an ATM/POS processor such that the plurality of ports are connected to a SONET network and the fragmentation process produces ATM cells, as shown by Jha. This would enable the high bandwidth SONET frames to transport Ethernet over ATM using standard protocols, as shown by Jha (Fig. 2; Col. 2, lines 35-41).

- In regards to Claim 23,

Cam discloses a packet data transfer process that covers all limitations of the parent claim.

Cam does not explicitly show monitoring and determining if the data packet has been fully fragmented.

Bucholz discloses a packet delivery system in which packets are fragmented for transmission (Title; Abstract). Referring to Fig. 6, Bucholz shows that a reassembly header is stored in the fragmented packet indicating its place within the packet, total packet length, total fragments, etc. such that it can be determined when the packet is fully fragmented (Col. 6-7, lines 63-23; claim 23 – monitoring and determining if the data packet has been fully fragmented).

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify the method of Cam by storing a reassembly header for the packet currently being processed, including information regarding how much of the packet has been fragmented and how much remains, so that subsequent processing of the remainder of the packet fragments can be performed, as taught by Bucholz. This would enable the transmission system to recognize when a complete packet has been processed when transmitted in a number of fragments.

5. Claim 25 is rejected under 35 U.S.C. 103(a) as being unpatentable over Cam in view of Bucholz as applied to claim 24 above, and further in view of Muller et al. (US006105122A), hereafter Muller.

- In regards to Claim 25,

Cam discloses a packet data transfer process that covers all limitations of the parent claim.

Cam shows that PHY device addresses may be segmented to achieve balance between efficient memory mapping and address decoding. Cam does not explicitly disclose an unbalanced port-loading condition as a port-switching event.

Muller discloses a switch configuration in a multi-node processing system that directs transmission messages between nodes in order to balance the load of the network (Col. 26-27, lines 61-5; claim 25 – port-switching event is an unbalanced port-loading condition).

It would have been obvious to one of ordinary skill in the art at the time of the invention to modify the process of Cam by considering an unbalanced port-loading condition as a port-switching event, as shown by Muller. This would further prevent any one port in Cam from consuming a disproportionate amount of bandwidth from the other ports and improve network utilization and efficiency.

(10) Response to Argument

- On pgs. 11-12 of the Brief, Appellant contends that neither Cam nor Bucholz discloses "fragmenting a first portion of a first available data packet into at least one data cell having a defined size, wherein this fragmentation continues until a user-defined number of cells is generated" as in claim 1, 9, 24, 26, and 32. Applicant further alleges that Cam does not even relate to packet fragmentation, but rather, to limiting the size of a data transfer.
- The Examiner respectfully disagrees. Cam does pertain to limiting the size of a data transfer, as stated by Appellant. However, the portion of Cam cited by Appellant on pg. 12 of the Brief shows that Cam performs this limiting of data transfer size by fragmenting packets into bounded block(s) of a set maximum size. Therefore, contrary to Appellants assertion, Cam does relate to packet fragmentation, and meets the contested claim limitations, as shown. It is noted that the disclosure of

Cam is not required to pertain exclusively, or even primarily, to packet fragmentation in order to properly read upon Appellant's claim limitations.

- On pgs. 12-14 of the Brief, Appellant contends that the reassembly header disclosed by Bucholz does not read upon the claimed step of "storing a data element that enables fragmenting of a second portion of a first available data packet subsequent to fragmenting at least a portion of a second available packet" as in claims 1, 9, 21, 24, 29, 32, and 42. Appellant alleges that Bucholz reassembly header, unlike Appellant's data element, does not provide information to enable fragmenting of a second portion subsequent to fragmenting at least a portion of a second available packet.
- The Examiner respectfully disagrees. It is necessary to consider the reassembly header of Bucholz *in combination* (emphasis added) with Cam in the claim rejections. As shown in the rejection, Figs. 5, 6, and 11 of Cam illustrates how multiple PHY devices transfer a maximum block size of a packet at a time, at which point a next PHY device is allowed to transmit. When all PHY devices have been given an opportunity to transmit, the initial PHY is permitted to continue fragmentation to transfer another maximum block size of a pending packet. Cam does not explicitly disclose how the PHY tracks the amount of a packet that has been transmitted and the amount remaining to transmit. Bucholz' reassembly

header is thus relied upon to illustrate how a stored data element (the header) can be used to track the fragmentation progress of a multi-fragment packet transmission in the multiple PHY environment of Cam. Bucholz' header would provide a way of "enabling" the PHY devices in Cam to continue fragmentation and transmission of the pending packet at the PHY's next transmission time, thereby meeting the contested claim limitations. It is noted that the claim limitations contested by Appellant contain broad language that reads upon various implementations of "enabling" fragmentation of a packet.

- On pgs. 14-15 of the Brief, Appellant contends that Cam does not disclose "monitoring the number of cells produced to determine if a user defined number are generated", as in claims 2, 10, and 27, because Cam does not relate to packet fragmentation.
- The Examiner respectfully disagrees. As shown in response to the first argument of the Brief, Cam is shown to relate to packet fragmentation as a way of limiting the size of data transfer. Further, as also shown in the disclosure of Cam cited by Appellant on pg. 12 of the Brief, Cam discloses that a fragment of a packet up to a maximum block size may be transferred, where this maximum block size is defined by a user, thereby meeting this limitation of the claims.

- On pgs. 15-16 of the Brief, Appellant contends that the reassembly header of Bucholz does not properly "enable subsequent processing of the remainder of the data packet currently being processed", as in claim 4.
- The Examiner respectfully disagrees. As previously argued in the Response to Arguments section of the Office Action filed 7/17/2006, the rejection of the contested claim depends upon the multiple fields of the reassembly header in combination with each other, rather than any one field. Appellant has shown how each individual field of Bucholz' header is not "indicative of the incomplete fragmentation status of the first available packet", but has not considered how these fields, when taken together within the common header (data element) provides an indication of the incomplete fragmentation status of the packet that enables subsequent processing of the remainder of the data packet currently being processed. In particular, the Total Fragment Field 640 and Fragment Number Field 650, when taken together, provide an indication of incomplete fragmentation status. As a packet fragment is processed for transmission, the Total Fragment Field 640 of the header attached to that packet fragment having a value X and the Fragment Number Field 650 having a value Y that is less than X would provide an indication that the packet contains further data to be fragmented, thus meeting the contested claim limitations.

- On pg. 17-18 of the Brief, Appellant alleges that Colmant does not disclose "initiating fragmentation on a data packet from another port while the fragmentation of the data packet on a first port continues until the user-defined number of cells is generated" as in claims 5, 8, 14, and 17. Appellant contends that the disclosure of Colmant does not relate to the same manner of packet fragmentation considered by the claims.
- The Examiner respectfully disagrees. Colmant is not required to disclose each and every limitation of the packet fragmentation claimed by Appellant in order to be applicable in rejecting the claims in combination with Cam and/or Bucholz. As admitted by Appellant, Colmant discloses packet fragmentation in a multi-port communication system, and is therefore analogous art for the purposes of combining with Cam and Bucholz to render obvious the claim limitations contested by Appellant. As shown in the rejection, Figs. 1B and 3B of Colmant illustrates how packets received on multiple ports can be fragmented at the same time, thereby improving medium utilization and keeping waiting time for a blocked port to a minimum. Therefore, the combination of this disclosure with the methods and processes disclosed by Cam and Bucholz properly meets the limitations, as claimed.

- On pg. 18-19 of the Brief, Appellant contends that the rejection of claims 20-23 and 43 is an exercise in improper hindsight reconstruction. Appellant alleges the rejection merely lists certain components of Appellant's invention without providing a motivation to combine the references. Appellant continues that simply showing that the ATM protocol existed at the time of the invention is not enough to satisfy the requirement of providing motivation.
- The Examiner respectfully disagrees. As shown in the above claim rejections, Cam discloses a method and process of using packet fragmentation to limit the size of data transfer in a network. Cam explicitly considers the applicability of such a process in an ATM network, as shown in the disclosure cited by Appellant on pg. 12 of the Brief. Jha is relied upon to show that ATM communication can be incorporated within other types of networks, such as SONET/SDH, IP and POS. The combination of Jha and Cam illustrate that the packet fragmentation taught by Cam can be applicable and beneficial to various networks. The ability to provide services in ATM over SONET and other standard protocols provides the motivation for applying Cam to the specific networks disclosed by Jha, as shown in the rejection. The rejection does not rely on Appellant's disclosure for motivation, as alleged by Appellant.

- On pg. 19-20 of the Brief, Appellant contends that the reliance on Muller in the in disclosing a "port-switching event is an unbalanced port-loading condition" is improper because Muller pertains to a switch configuration that does not relate to packet fragmentation.
- The Examiner respectfully disagrees. Muller discloses switching data between multiple ports in a multi-processor system. Furthermore, Muller discloses fragmented PIT packets within the disclosed system (Col. 36, line 50). Therefore, Muller qualifies as analogous art for combining with the disclosure of Cam. As shown in the rejection, Muller discloses directing transmission messages (port-switching events) between nodes in order to balance the load of the network. Combining this teaching of Muller with Cam would prevent any one port (PHY) from consuming a disproportionate amount of bandwidth, thereby meeting the contested claim limitations.

(11) Related Proceeding(s) Appendix

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

Gregory Sefcheck

Gregory Sefcheck

6/14/2007

Wing Chan 6/22/07

WING CHAN
SUPERVISORY PATENT EXAMINER

Conferees:

Wing Chan

Seema Rao

Seema S. Rao

SEEMA S. RAO 6/18/07
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2600